

# Positive-Unlabeled Learning with Non-Negative Risk Estimator

Ryuichi Kiryo<sup>1</sup> Gang Niu<sup>1</sup> Marthinus C. du Plessis<sup>1</sup> Masashi Sugiyama<sup>2,1</sup>

## Abstract

From only *positive* (P) and *unlabeled* (U) data, a binary classifier could be trained with PU learning. Unbiased PU learning that is based on *unbiased risk estimators* is now state of the art. However, if its model is very flexible, its empirical risk on training data will go negative, and we will suffer from overfitting seriously. In this paper, we propose a novel *non-negative risk estimator* for PU learning. When being minimized, it is more robust against overfitting, and thus we are able to train very flexible models given limited P data. Moreover, we analyze the *bias*, *consistency* and *mean-squared-error reduction* of the proposed risk estimator as well as the *estimation error* of the corresponding risk minimizer. Experiments show that the non-negative risk estimator outperforms unbiased counterparts when they disagree.

## 1. Introduction

Positive-unlabeled (PU) learning can be dated back to Denis (1998); De Comité et al. (1999); Letouzey et al. (2000) and has been well known since then. It mainly focuses on binary classification with application to retrieval and novelty or outlier detection (Elkan & Noto, 2008; Ward et al., 2009; Scott & Blanchard, 2009; Blanchard et al., 2010), while it also has applications in matrix completion (Hsieh et al., 2015) and sequential data (Li et al., 2009; Nguyen et al., 2011).

Existing PU methods for binary classification could be divided into two categories based on how U data is handled. The first category (e.g., Liu et al., 2002; Li & Liu, 2003) naively identifies U data that are likely to be *negative* (N), and then runs PN learning (i.e., ordinary supervised learning). The second (e.g., Lee & Liu, 2003; Liu et al., 2003) regards all U data as N data but they are weighted smaller. The former relies on the heuristics for identifying possible

N data too much, and it is found inferior to the latter (Liu et al., 2003); the latter also needs to tune the weights if not set heuristically, which is computationally expensive.

In order to avoid tuning the weights, unbiased PU learning comes into play as a subcategory of the second category. A milestone is Elkan & Noto (2008), which regards a U data as weighted P and N data simultaneously. It might lead to *unbiased risk estimators*, if we unrealistically assume that the class-posterior probability is one for all P data.<sup>1</sup> A breakthrough in this direction is du Plessis et al. (2014), for proposing the first unbiased risk estimator, and a more general estimator was suggested in du Plessis et al. (2015) as a common foundation of them. The former is unbiased when the risk is w.r.t. some symmetric losses; the latter is always unbiased, and it is convex w.r.t. some linear-odd losses. In the latter, there are three partial risks in the total risk (cf. Eq. (3) defined later): Besides the two shared by the second category, it has another *negative risk* viewing P data as N data to cancel the bias from viewing U data as N data. PU learning with this estimator is the current state of the art.

Nevertheless, that estimator will give a negative empirical risk, if the model being trained is very flexible. The worst is the model can become any measurable function and the loss is not upper bounded, so that the empirical risk is not lower bounded. This makes no sense since the *risk*, which is the object to be estimated, could never be negative.

In this paper, we propose a novel *non-negative risk estimator* that follows and improves on the unbiased risk estimator mentioned above. It can be used for two purposes:

- Given some validation data (which are also PU data), we can use it to evaluate the risk. For this case, it is a *biased yet optimal* estimator; for some symmetric losses, the *reduction in mean squared error* is guaranteed;
- We can also use it to train a binary classifier. For this case, its *estimation error bound* has the same order as unbiased counterparts (cf. Niu et al., 2016). In experiments, it compares favorably with those counterparts.

<sup>1</sup>The University of Tokyo, Tokyo, Japan <sup>2</sup>RIKEN, Tokyo, Japan. Correspondence to: Gang Niu <gang@ms.k.u-tokyo.ac.jp>, Masashi Sugiyama <sugi@k.u-tokyo.ac.jp>.

<sup>1</sup>It implies the P and N class-conditional densities have disjoint support sets and then any P and N data can be perfectly separated by a fixed classifier.

In addition, we design a *large-scale* PU learning algorithm that can minimize both the unbiased and non-negative risk estimators. This algorithm can use any *surrogate loss*, and it is based on *stochastic optimization* (e.g., Kingma & Ba, 2015). Note that Sansone et al. (2016) is the only existing large-scale PU algorithm, but it could use only a surrogate loss in du Plessis et al. (2015), and it is basically based on *sequential minimal optimization* (Platt, 1999).

The rest of this paper is organized as follows. In Section 2 we review unbiased PU learning. In Section 3 we propose non-negative PU learning. Theoretical analyses are carried out in Section 4. Experiments are discussed in Section 5.

## 2. Unbiased PU learning

In this section, we review unbiased PU learning (du Plessis et al., 2014; 2015).

**Problem settings** Let  $X \in \mathbb{R}^d$  and  $Y \in \{\pm 1\}$  (where  $d$  is a natural number) be the input and output random variables. Let  $p(x, y)$  be the *underlying joint density*,

$$p_p(x) = p(x | Y = +1), \quad p_n(x) = p(x | Y = -1)$$

be the *P* and *N* marginals (i.e., the class-conditional densities),  $p(x)$  be the *U* marginal, and  $\pi_p = p(Y = +1)$  be the *class-prior probability* with  $\pi_n = 1 - \pi_p$ . We assume that  $\pi_p$  is known throughout the paper, while it could be effectively estimated from only P and U data (see, for example, du Plessis et al., 2017; Ramaswamy et al., 2016).

Consider the two-sample problem setting of PU learning (Ward et al., 2009): Two sets of data are sampled independently from  $p_p(x)$  and  $p_n(x)$  as

$$\mathcal{X}_p = \{x_i^p\}_{i=1}^{n_p} \stackrel{\text{i.i.d.}}{\sim} p_p(x), \quad \mathcal{X}_u = \{x_i^u\}_{i=1}^{n_u} \stackrel{\text{i.i.d.}}{\sim} p(x),$$

and a classifier needs to be trained from  $\mathcal{X}_p$  and  $\mathcal{X}_u$ . If it is PN learning,  $\mathcal{X}_n = \{x_i^n\}_{i=1}^{n_n} \stackrel{\text{i.i.d.}}{\sim} p_n(x)$  rather than  $\mathcal{X}_u$  will be available so that a classifier can be trained from  $\mathcal{X}_p$  and  $\mathcal{X}_n$  by many supervised learning methods.

**Risk estimators** Unbiased PU learning entirely relies on unbiased estimators to the risk. Let  $g : \mathbb{R}^d \rightarrow \mathbb{R}$  be a *decision function* for binary classification, and  $\ell : \mathbb{R} \times \{\pm 1\} \rightarrow \mathbb{R}$  be a *loss function*. Denote by

$$R_p^+(g) = \mathbb{E}_p[\ell(g(X), +1)], \quad R_n(g) = \mathbb{E}_n[\ell(g(X), -1)],$$

where  $\mathbb{E}_p[\cdot] = \mathbb{E}_{X \sim p_p}[\cdot]$  and  $\mathbb{E}_n[\cdot] = \mathbb{E}_{X \sim p_n}[\cdot]$ . Then, the *risk of g*, namely  $R(g) = \mathbb{E}_{(X,Y)}[\ell(g(X), Y)]$ , is given by

$$R(g) = \pi_p R_p^+(g) + \pi_n R_n(g). \quad (1)$$

In PN learning,  $R(g)$  can be approximated by

$$\hat{R}_{pn}(g) = \pi_p \hat{R}_p^+(g) + \pi_n \hat{R}_n(g), \quad (2)$$

where  $\hat{R}_p^+(g) = \frac{1}{n_p} \sum_{i=1}^{n_p} \ell(g(x_i^p), +1)$  and  $\hat{R}_n(g) = \frac{1}{n_n} \sum_{i=1}^{n_n} \ell(g(x_i^n), -1)$ . In PU learning,  $\mathcal{X}_n$  is unavailable, so that  $R_n(g)$  must be approximated indirectly (du Plessis et al., 2014; 2015): Denote by

$$R_p^-(g) = \mathbb{E}_p[\ell(g(X), -1)], \quad R_u^-(g) = \mathbb{E}_X[\ell(g(X), -1)],$$

then  $\pi_n R_n(g) = R_u^-(g) - \pi_p R_p^-(g)$  as  $\pi_n p_n(x) = p(x) - \pi_p p_p(x)$ , and hence we obtain

$$\hat{R}_{pu}(g) = \pi_p \hat{R}_p^+(g) - \pi_p \hat{R}_p^-(g) + \hat{R}_u^-(g), \quad (3)$$

where  $\hat{R}_p^-(g)$  and  $\hat{R}_u^-(g)$  are the empirical averages corresponding to  $R_p^-(g)$  and  $R_u^-(g)$ .

The *empirical risk estimators*  $\hat{R}_{pn}(g)$  and  $\hat{R}_{pu}(g)$  are both *unbiased and consistent*.<sup>2</sup> When they are used for evaluation and cross-validation, the loss  $\ell$  is by default the *zero-one loss*  $\ell_{01}(t, y) = (1 - \text{sign}(ty))/2$  that is non-smooth. When being used for training,  $\ell_{01}$  is replaced with a *surrogate loss* that is usually smooth, e.g., Lipschitz-continuous or even differentiable. In particular, du Plessis et al. (2014) showed that if  $\ell$  satisfies a *symmetric condition*:

$$\ell(t, +1) + \ell(t, -1) = 1, \quad (4)$$

we will have

$$\hat{R}_{pu}(g) = 2\pi_p \hat{R}_p^+(g) + \hat{R}_u^-(g) - \pi_p, \quad (5)$$

which can be optimized by separating  $\mathcal{X}_p$  and  $\mathcal{X}_u$  with off-the-shelf methods of cost-sensitive learning. An issue is  $\hat{R}_{pu}(g)$  in (5) must be non-convex since no  $\ell$  in (4) can be convex. To this end, du Plessis et al. (2015) showed that  $\hat{R}_{pu}(g)$  in (3) will be convex, if  $\ell$  is convex in  $t$  and meets a *linear-odd condition* (cf. Patrini et al., 2016):

$$\ell(t, +1) - \ell(t, -1) = -t. \quad (6)$$

This results in a convex optimization problem so long as  $g$  is linear in its parameters,<sup>3</sup> for which the globally optimal solution can be obtained. Note that (6) is also a necessary condition, if  $\ell$  is in fact unary, i.e.,  $\ell(t, -1) = \ell(-t, +1)$ .

**Justifications** Thanks to the unbiasedness, we can study theoretical guarantees when learning is involved. Let  $\mathcal{G}$  be the *function class* from which the decision function can be selected, and  $\hat{g}_{pn}$  and  $\hat{g}_{pu}$  be the *empirical risk minimizers* to  $\hat{R}_{pn}(g)$  and  $\hat{R}_{pu}(g)$ . Niu et al. (2016) showed that if

- the loss  $\ell$  satisfies (4) and is *Lipschitz-continuous*;

<sup>2</sup>The consistency here means that for any fixed  $g$ ,  $\hat{R}_{pn}(g)$  and  $\hat{R}_{pu}(g)$  converge to  $R(g)$  as  $n_p$ ,  $n_n$  and  $n_u$  approach infinity.

<sup>3</sup>If  $g(x; \theta)$  is convex but non-linear in  $\theta$ ,  $\hat{R}_p^+(g) - \hat{R}_p^-(g) = -(1/n_p) \sum_{i=1}^{n_p} g(x_i^p; \theta)$  will be concave in  $\theta$ .

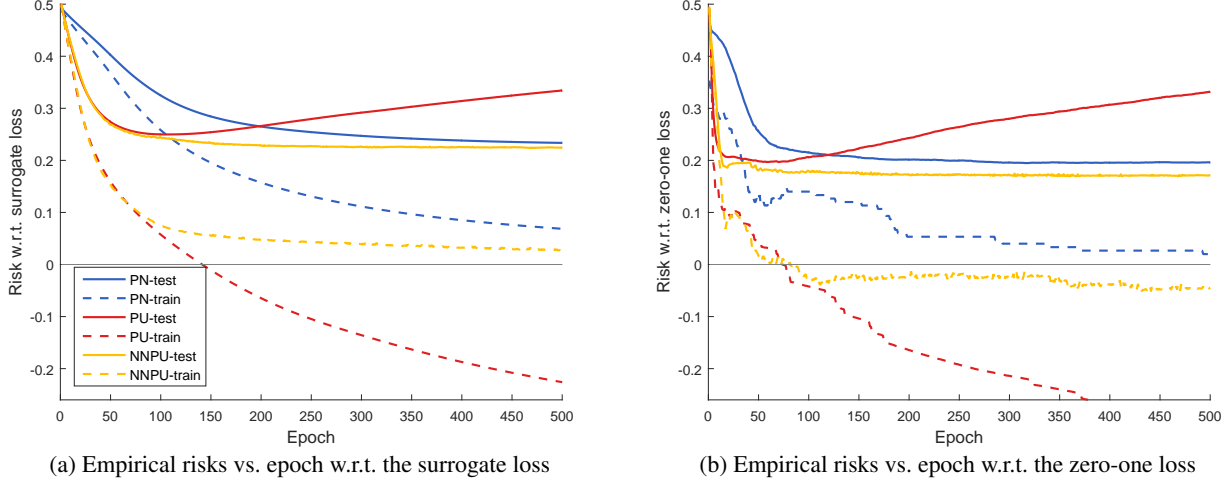


Figure 1. Illustrative experimental results of PN, unbiased PU and non-negative PU (NNPU) learning. The dataset is MNIST; even/odd digits are regarded as the P/N class and  $\pi_p \approx 1/2$ ; for PN learning  $n_p = 100$  and  $n_n = 50$  while for PU and NNPU learning  $n_p = 100$  and  $n_u = 59,900$ . The model is a multilayer perceptron (784-100-1) with ReLU activations (trained by stochastic optimization). Solid curves are  $\hat{R}_{pn}(g)$  on test data where  $g \in \{\hat{g}_{pn}, \hat{g}_{pu}, \tilde{g}_{pu}\}$ , and dashed curves are  $\hat{R}_{pn}(\hat{g}_{pn})$ ,  $\hat{R}_{pu}(\hat{g}_{pu})$  and  $\hat{R}_{pu}(\tilde{g}_{pu})$  on training data.

- the Rademacher complexity of  $\mathcal{G}$  decays in  $\mathcal{O}(1/\sqrt{n})$  for data of size  $n$  from  $p(x)$ ,  $p_p(x)$ , or  $p_n(x)$ ,<sup>4</sup>

then, the estimation error bound of  $\hat{g}_{pu}$  is tighter than that of  $\hat{g}_{pn}$  whenever  $\pi_p/\sqrt{n_p} + 1/\sqrt{n_u} < \pi_n/\sqrt{n_n}$ . In other words, given these conditions, PU learning is likely to outperform PN learning.

### 3. Non-negative PU learning

In this section, we introduce a non-negative risk estimator. When being minimized, it is more robust against the overfitting problem than those unbiased counterparts.

#### 3.1. Motivation

Let us look inside the aforementioned justification of unbiased PU learning. Intuitively, the advantage is solely from the transformation  $\pi_n R_n(g) = R_u^-(g) - \pi_p R_p^-(g)$ : When we approximate  $\pi_n R_n(g)$  directly, the convergence rate is  $\mathcal{O}_p(\pi_n/\sqrt{n_n})$ , where  $\mathcal{O}_p$  denotes the order in probability; when we approximate it indirectly with the right-hand side above, the convergence rate is  $\mathcal{O}_p(\pi_p/\sqrt{n_p} + 1/\sqrt{n_u})$ . Therefore, we can benefit from a smaller *uniform deviation* when  $n_p$  and  $n_u$  dominate  $n_n$ .

However, the critical assumption on the Rademacher complexity is indispensable, otherwise it will be more difficult for the estimation error bound of  $\hat{g}_{pu}$  to be tighter than that of  $\hat{g}_{pn}$ . Moreover, as the complexity decays slower, those bounds become looser, and then we are not sure that  $\hat{g}_{pu}$  is

<sup>4</sup>The Rademacher complexity of  $\mathcal{G}$  for  $\mathcal{X} \sim q(x)$  of size  $n$  is defined by  $\mathfrak{R}_{n,q}(\mathcal{G}) = \mathbb{E}_{\mathcal{X}} \mathbb{E}_{\sigma_1, \dots, \sigma_n} [\sup_{g \in \mathcal{G}} \frac{1}{n} \sum_{i \in \mathcal{X}} \sigma_i g(x_i)]$  where each  $\sigma_i$  is a Rademacher variable (Mohri et al., 2012).

likely to be better even when its bound is tighter. Thus, in order to train  $\hat{g}_{pu}$  in practice,  $\mathcal{G}$  cannot be too complex, or equivalently the model for  $g$  cannot be too flexible.

This argument has been validated experimentally, and the illustrative experimental results are reported in Figure 1. A *multilayer perceptron* was trained for separating the even and odd digits of MNIST hand-written digits. The model is so flexible that the number of parameters is 500 times more than labeled data especially the P and N data come from several subclasses; furthermore, ReLU is unbounded which means we cannot bound the complexities tightly (cf. Theorem 18 in Bartlett & Mendelson, 2002, where the activations have a unit infinity norm). Consequently, we can see from Figure 1:

- On training data, the risks of  $\hat{g}_{pu}$  and  $\hat{g}_{pn}$  always decrease, and the former is much faster than the latter;
- On test data, the risk of  $\hat{g}_{pn}$  always decreases but that of  $\hat{g}_{pu}$  does not. The risk of  $\hat{g}_{pu}$  is much lower at the beginning but much higher at the end;
- Similar phenomena can also be observed for the empirical risks w.r.t. the zero-one loss.

To sum up,  $\hat{g}_{pu}$  is able to fit training data quickly; unfortunately, it becomes overfitting also quickly. This evidences that in order to train  $\hat{g}_{pu}$ , the model cannot be too flexible.

#### 3.2. Non-negative risk estimator

Nevertheless, sometimes we have no choice: We are interested in flexible models, while labeling more data is out of our control. Can we alleviate the overfitting problem with neither changing the model nor labeling more data?

Table 1. Loss functions for PU learning and their properties, all of which are unary such that  $\ell(t, y) = \ell(z)$  with  $z = ty$ . The ramp loss was from du Plessis et al. (2014); the double hinge loss was from du Plessis et al. (2015), where the squared, logistic, and hinge losses were also discussed. The ramp and squared losses are scaled in order to satisfy (4) or (6). Notice that the sigmoid loss is a horizontally mirrored logistic function, whereas the logistic loss is the negative logarithm of the logistic function.

Name	Definition	Satisfy (4)	Satisfy (6)	Bounded	Lipschitz	Non-zero sub-gradient
Zero-one loss	$(1 - \text{sign}(z))/2$	✓	×	✓	×	$z = 0$
Ramp loss	$\max\{0, \min\{1, (1 - z)/2\}\}$	✓	×	✓	✓	$z \in [-1, +1]$
Squared loss	$(z - 1)^2/4$	×	✓	×	×	$z \in \mathbb{R}$
Logistic loss	$\ln(1 + \exp(-z))$	×	✓	×	✓	$z \in \mathbb{R}$
Hinge loss	$\max\{0, 1 - z\}$	×	×	×	✓	$z \in (-\infty, +1]$
Double hinge loss	$\max\{0, (1 - z)/2, -z\}$	×	✓	×	✓	$z \in (-\infty, +1]$
Sigmoid loss	$1/(1 + \exp(z))$	✓	×	✓	✓	$z \in \mathbb{R}$

The answer is positive. An obvious reason for that overfitting problem can be identified:  $\hat{R}_{\text{pu}}(\hat{g}_{\text{pu}})$  keeps decreasing and becomes negative. This is stupid, as  $R(g)$  could never be negative. More specifically, it holds that

$$R_{\text{u}}^-(g) - \pi_{\text{p}} R_{\text{p}}^-(g) = \pi_{\text{n}} R_{\text{n}}(g) \geq 0,$$

but it is not guaranteed that  $\hat{R}_{\text{u}}^-(g) - \pi_{\text{p}} \hat{R}_{\text{p}}^-(g) \geq 0$  when we go from (1) to (3), whereas it is true that  $\pi_{\text{n}} \hat{R}_{\text{n}}(g) \geq 0$  when we go from (1) to (2). This might be the most likely reason for  $\hat{g}_{\text{pu}}$  to overfit as a minimizer to  $\hat{R}_{\text{pu}}(g)$ .

Based on this key observation, we propose a *non-negative risk estimator*:

$$\tilde{R}_{\text{pu}}(g) = \pi_{\text{p}} \hat{R}_{\text{p}}^+(g) + \max\{0, \hat{R}_{\text{u}}^-(g) - \pi_{\text{p}} \hat{R}_{\text{p}}^-(g)\}, \quad (7)$$

and let  $\tilde{g}_{\text{pu}}$  be the empirical risk minimizer to  $\tilde{R}_{\text{pu}}(g)$  in  $\mathcal{G}$  defined by  $\tilde{g}_{\text{pu}} = \arg \min_{g \in \mathcal{G}} \tilde{R}_{\text{pu}}(g)$ . We refer to training this  $\tilde{g}_{\text{pu}}$  as *non-negative PU (NNPU) learning*. The implementation of NNPU learning will be given in Section 3.3, and theoretical analyses of  $\tilde{R}_{\text{pu}}(g)$  and  $\tilde{g}_{\text{pu}}$  will be in Section 4. Again, we can see from Figure 1:

- On training data, the risk of  $\tilde{g}_{\text{pu}}$  first drops quickly as  $\hat{g}_{\text{pu}}$ , then becomes almost flat and does not further go down with  $\hat{g}_{\text{pu}}$ , so that the risk of  $\tilde{g}_{\text{pu}}$  is closer to the risk of  $\hat{g}_{\text{pn}}$  and farther from that of  $\hat{g}_{\text{pu}}$ ;
- On test data, the tendency is similar, and a difference is that the risk of  $\tilde{g}_{\text{pu}}$  does not go up with  $\hat{g}_{\text{pu}}$  instead of go down with it;
- At the end,  $\tilde{g}_{\text{pu}}$  possesses the lowest risk on test data w.r.t. either the surrogate loss or the zero-one loss.

In summary,  $\tilde{g}_{\text{pu}}$  successfully combines the advantages of  $\hat{g}_{\text{pn}}$  and  $\hat{g}_{\text{pu}}$ . It fits training data as quickly as  $\hat{g}_{\text{pu}}$  and thus more quickly than  $\hat{g}_{\text{pn}}$ ; it is as robust as  $\hat{g}_{\text{pn}}$  and thus more robust than  $\hat{g}_{\text{pu}}$  against overfitting.

### 3.3. Implementation

A list of popular loss functions and their properties is shown in Table 1. Let  $g$  be parameterized by  $\theta$ . Then, if

$g$  is linear in  $\theta$ , the losses satisfying (6) should be preferable for that they result in convex optimizations: (7) can be rewritten as  $\tilde{R}_{\text{pu}}(g) = \max\{\pi_{\text{p}} \hat{R}_{\text{p}}^+(g), \hat{R}_{\text{pu}}(g)\}$  that as the larger of two convex functions is convex. However, if  $g$  needs to be flexible, it is often non-linear in  $\theta$ . Then the losses satisfying (4) should be preferable, since the optimization is anyway non-convex and bounded losses lead to bounded risks that are easier to minimize.

In du Plessis et al. (2014), a scaled *ramp loss* was selected as the surrogate loss, and  $\hat{R}_{\text{pu}}(g)$  w.r.t. this loss was minimized by the *concave-convex procedure* (Yuille & Rangarajan, 2001). This solver is already fairly complicated, and the solver resulted from replacing  $\hat{R}_{\text{pu}}(g)$  with  $\tilde{R}_{\text{pu}}(g)$  will be even more difficult to implement. Therefore, we propose to use another surrogate loss, namely, the *sigmoid loss* defined by

$$\ell_{\text{sig}}(t, y) = 1/(1 + \exp(ty)),$$

which is a horizontally mirrored logistic function. Its advantage over the ramp loss is clear: The gradient is everywhere non-zero and empirical risks w.r.t. it could be minimized by off-the-shelf gradient methods.

In front of big data, we would like to scale PU learning up from batch to stochastic optimization. Minimizing  $\hat{R}_{\text{pu}}(g)$  is embarrassingly parallel while minimizing  $\tilde{R}_{\text{pu}}(g)$  is not since the former is point-wise while the latter is not due to the max in it. That being said, we have

$$\begin{aligned} & \max\{0, \hat{R}_{\text{u}}^-(g; \mathcal{X}_{\text{u}}) - \pi_{\text{p}} \hat{R}_{\text{p}}^-(g; \mathcal{X}_{\text{p}})\} \\ &= \max\{0, (1/N) \sum_{i=1}^N (\hat{R}_{\text{u}}^-(g; \mathcal{X}_{\text{u}}^i) - \pi_{\text{p}} \hat{R}_{\text{p}}^-(g; \mathcal{X}_{\text{p}}^i))\} \\ &\leq (1/N) \sum_{i=1}^N \max\{0, \hat{R}_{\text{u}}^-(g; \mathcal{X}_{\text{u}}^i) - \pi_{\text{p}} \hat{R}_{\text{p}}^-(g; \mathcal{X}_{\text{p}}^i)\}, \end{aligned}$$

where training data  $(\mathcal{X}_{\text{p}}, \mathcal{X}_{\text{u}})$  are partitioned into  $N$  mini-batches of size  $(n_{\text{p}}/N, n_{\text{u}}/N)$ . Hence, an upper bound of  $\tilde{R}_{\text{pu}}(g)$  can be minimized in parallel.

Algorithm 1 describes large-scale PU learning. It has two hyperparameters  $0 \leq \beta \leq \pi_{\text{p}} \sup_{t,y} \ell(t, y)$  and  $0 \leq \gamma \leq 1$  (excluding those of  $\mathcal{A}$ ). In practice, if  $\hat{R}_{\text{u}}^-(g; \mathcal{X}_{\text{u}}^i) -$



**Algorithm 1** Large-scale PU learning

**Input:** training data  $(\mathcal{X}_p, \mathcal{X}_u)$ , hyperparameters  $\beta, \gamma$  such that  $0 \leq \beta \leq \pi_p \sup_{t,y} \ell(t, y)$  and  $0 \leq \gamma \leq 1$   
**Output:** model parameter  $\theta$  for  $\hat{g}_{pu}(x; \theta)$  or  $\tilde{g}_{pu}(x; \theta)$

- 1: Let  $\mathcal{A}$  be an external SGD-like algorithm;
- 2: **while** no stopping criterion has been met **do**
- 3:   Shuffle the data into  $N$  mini-batches, and denote by  $(\mathcal{X}_p^i, \mathcal{X}_u^i)$  the  $i$ -th mini-batch;
- 4:   **for**  $i = 1$  **to**  $N$  **do**
- 5:     **if**  $\hat{R}_u^-(g; \mathcal{X}_u^i) - \pi_p \hat{R}_p^-(g; \mathcal{X}_p^i) \geq -\beta$  **then**
- 6:       Set gradient  $\nabla_{\theta} \hat{R}_{pu}(g; \mathcal{X}_p^i, \mathcal{X}_u^i)$ ;
- 7:       Update  $\theta$  by  $\mathcal{A}$  with its current step size  $\eta$ ;
- 8:     **else**
- 9:       Set gradient  $\nabla_{\theta} (\pi_p \hat{R}_p^-(g; \mathcal{X}_p^i) - \hat{R}_u^-(g; \mathcal{X}_u^i))$ ;
- 10:      Update  $\theta$  by  $\mathcal{A}$  but with step size  $\gamma\eta$ ;
- 11:    **end if**
- 12:   **end for**
- 13: **end while**

$\pi_p \hat{R}_p^-(g; \mathcal{X}_p^i)$  is slightly negative, we may tolerate it by minimizing

$$\pi_p \hat{R}_p^+(g; \mathcal{X}_p^i) + \max\{-\beta, \hat{R}_u^-(g; \mathcal{X}_u^i) - \pi_p \hat{R}_p^-(g; \mathcal{X}_p^i)\}$$

in every mini-batch, otherwise the upper bound of  $\tilde{R}_{pu}(g)$  may be very loose. The hyperparameter  $\beta$  controls the degree of our tolerance. If  $\beta = 0$ , we are minimizing exactly the upper bound of  $\tilde{R}_{pu}(g)$ , and if  $\beta$  is large, we are minimizing  $\hat{R}_{pu}(g)$ . Additionally, there is an algorithmic trick when  $\hat{R}_u^-(g; \mathcal{X}_u^i) - \pi_p \hat{R}_p^-(g; \mathcal{X}_p^i) < -\beta$ : Instead of going down, we go up along  $\nabla_{\theta} (\hat{R}_u^-(g; \mathcal{X}_u^i) - \pi_p \hat{R}_p^-(g; \mathcal{X}_p^i))$  to make the  $i$ -th mini-batch less overfitted. The hyperparameter  $\gamma$  is the discount factor to the step size and it has to be tuned for stability. If  $\mathcal{A}$  has adaptive updates such as *Adagrad* (Duchi et al., 2011) and *Adam* (Kingma & Ba, 2015), the algorithm will be insensitive to the choice of  $\gamma$ .

In Figure 1,  $\hat{g}_{pu}$  and  $\tilde{g}_{pu}$  were both trained by Algorithm 1 where  $\ell$  is  $\ell_{sig}$ ,  $\beta = 1/2$  for  $\hat{g}_{pu}$  and  $\beta = 0$  for  $\tilde{g}_{pu}$ , and  $\mathcal{A}$  is Adam with  $\gamma = 1$ .

## 4. Theoretical analyses

In this section, we analyze  $\tilde{R}_{pu}(g)$  and  $\tilde{g}_{pu}$ . The proofs of all the theoretical results can be found in Appendix A.

### 4.1. Bias and consistency

Note that for fixed  $g$  and any  $\mathcal{X}_p$  and  $\mathcal{X}_u$ ,  $\tilde{R}_{pu}(g) \geq \hat{R}_{pu}(g)$  but  $\hat{R}_{pu}(g)$  is unbiased, which means  $\tilde{R}_{pu}(g)$  is potentially biased. The most fundamental question then is whether or not  $\tilde{R}_{pu}(g)$  is consistent. From now on, we prove the consistency of  $\tilde{R}_{pu}(g)$ .

To begin with, let us partition all possible  $\mathcal{X}_p$  and  $\mathcal{X}_u$  into two sets depending on  $g$ :

$$\begin{aligned} \mathfrak{D}^+(g) &= \{(\mathcal{X}_p, \mathcal{X}_u) \mid \hat{R}_u^-(g) - \pi_p \hat{R}_p^-(g) \geq 0\}, \\ \mathfrak{D}^-(g) &= \{(\mathcal{X}_p, \mathcal{X}_u) \mid \hat{R}_u^-(g) - \pi_p \hat{R}_p^-(g) < 0\}. \end{aligned}$$

Without loss of generality, assume that  $\ell$  is bounded by  $C_{\ell}$ , i.e.,  $\sup_t \ell(t, \pm 1) \leq C_{\ell}$ ; if  $\ell$  is not globally bounded such as the squared loss, it may be locally bounded if  $t = g(x)$  where  $g \in \mathcal{G}$  by assuming  $\sup_{g \in \mathcal{G}} \|g\|_{\infty}$  is finite.

**Lemma 1.** *The following statements are equivalent:*

1. *The measure of  $\mathfrak{D}^-(g)$  is non-zero;*
2.  *$\tilde{R}_{pu}(g)$  differs from  $\hat{R}_{pu}(g)$  with a non-zero probability over repeated sampling of  $\mathcal{X}_p$  and  $\mathcal{X}_u$ ;*
3. *The bias of  $\tilde{R}_{pu}(g)$  is positive.*

In addition, assume that  $R_n(g) \geq \alpha > 0$ , and the measure of  $\mathfrak{D}^-(g)$  can be bounded by

$$\Pr(\mathfrak{D}^-(g)) \leq \exp\left(-\frac{2\alpha^2/C_{\ell}^2}{\pi_p^2/n_p + 1/n_u}\right). \quad (8)$$

Based on Lemma 1, we can show the exponential decay of the bias and also the consistency. For convenience, denote by  $\chi_{n_p, n_u} = 2\pi_p/\sqrt{n_p} + 1/\sqrt{n_u}$ .

**Theorem 2** (Bias and consistency). *Assume that  $R_n(g) \geq \alpha > 0$  and denote by  $\Delta_g$  the right-hand side of Eq. (8). As  $n_p, n_u \rightarrow \infty$ , the bias of  $\tilde{R}_{pu}(g)$  decays exponentially:*

$$\mathbb{E}_{\mathcal{X}_p, \mathcal{X}_u}[\tilde{R}_{pu}(g)] - R(g) \leq C_{\ell} \pi_p \Delta_g. \quad (9)$$

Moreover, for any  $\delta > 0$ , we have with probability at least  $1 - \delta$ ,

$$|\tilde{R}_{pu}(g) - R(g)| \leq C_{\delta} \cdot \chi_{n_p, n_u} + C_{\ell} \pi_p \Delta_g, \quad (10)$$

and with probability at least  $1 - \delta - \Delta_g$ ,

$$|\tilde{R}_{pu}(g) - R(g)| \leq C_{\delta} \cdot \chi_{n_p, n_u}, \quad (11)$$

where  $C_{\delta} = C_{\ell} \sqrt{\ln(2/\delta)/2}$ .

Either (10) or (11) in Theorem 2 indicates for any fixed  $g$ ,  $\tilde{R}_{pu}(g)$  converges to  $R(g)$  in  $\mathcal{O}_p(\pi_p/\sqrt{n_p} + 1/\sqrt{n_u})$ . In fact, this convergence rate is optimal in estimating a given expectation from data (Chung, 1968), which means  $\tilde{R}_{pu}(g)$  is a biased yet optimal estimator to the risk.

### 4.2. Mean squared error

After we introduce the bias,  $\tilde{R}_{pu}(g)$  tends to overestimate  $R(g)$ . It is not a shrinkage estimator and its mean squared error (MSE) is not necessarily smaller than that of  $\hat{R}_{pu}(g)$ . However, we are sure of this reduction for some special  $\ell$ .

**Theorem 3** (MSE reduction). *Assume that*

1.  $\Pr(\mathcal{D}^-(g)) > 0$ ;
2. the loss  $\ell$  satisfies Eq. (4);
3.  $R_u(g) \geq \alpha > 0$ ;
4.  $n_u$  is significantly larger than  $n_p$ , such that  $R_u^-(g) - \hat{R}_u^-(g) \leq 2\alpha$  almost surely on  $\mathcal{D}^-(g)$ .

Then,  $\text{MSE}(\tilde{R}_{pu}(g)) < \text{MSE}(\hat{R}_{pu}(g))$ .<sup>5</sup> Furthermore, for any  $0 \leq \beta \leq C_\ell \pi_p$ , it holds that

$$\begin{aligned} \text{MSE}(\hat{R}_{pu}(g)) - \text{MSE}(\tilde{R}_{pu}(g)) \\ \geq 3\beta^2 \Pr\{\tilde{R}_{pu}(g) - \hat{R}_{pu}(g) > \beta\}. \end{aligned} \quad (12)$$

The explanation of the fourth assumption in Theorem 3 is as follows. It is natural to assume  $n_u$  is significantly larger than and also grows faster than  $n_p$ , since  $U$  data are much cheaper than  $P$  data. Hence, it holds asymptotically that<sup>6</sup>

$$\frac{\Pr\{R_u^-(g) - \hat{R}_u^-(g) \geq \alpha\}}{\Pr\{\hat{R}_p^-(g) - R_p^-(g) \geq \alpha/\pi_p\}} \propto \exp(n_p - n_u),$$

which means compared with  $\mathcal{X}_p$ , the contribution of  $\mathcal{X}_u$  is negligible for making  $(\mathcal{X}_p, \mathcal{X}_u) \in \mathcal{D}^-(g)$ . As  $\Pr(\mathcal{D}^-(g))$  exhibits exponential decay mainly in  $n_p$  and  $\Pr\{R_u^-(g) - \hat{R}_u^-(g) \geq 2\alpha\}$  exhibits exponential decay in  $n_u$ , we could assume  $R_u^-(g) - \hat{R}_u^-(g) \leq 2\alpha$  almost surely on  $\mathcal{D}^-(g)$ .

Theorem 3 is not a necessary condition even for  $\ell$  meeting (4). In the proof, the reduction in MSE was expressed as a *Lebesgue-Stieltjes integral* (Halmos, 1974) that is intractable. To make the integral positive, we found a very simple sufficient condition under which the integrand itself is positive, but this is too strict. MSE reductions for other  $\ell$  have been observed experimentally; it is an open problem to prove them mathematically.

### 4.3. Estimation error

As a risk estimator,  $\tilde{R}_{pu}(g)$  can be used for either evaluating  $R(g)$  or training  $g$ . While Theorems 2 and 3 addressed the first purpose of evaluating  $R(g)$ , we are likewise interested in any theoretical guarantee for the second purpose of training  $g$ . In what follows, we analyze the estimation error  $R(\tilde{g}_{pu}) - R(g^*)$ , where  $g^*$  is the true risk minimizer in  $\mathcal{G}$ , i.e.,  $g^* = \arg \min_{g \in \mathcal{G}} R(g)$ .

Similarly, assume that  $\ell$  is Lipschitz-continuous in its first parameter with a Lipschitz constant  $L_\ell$ ; if  $\ell$  is not globally Lipschitz such as the squared loss, it may be locally Lipschitz if  $t = g(x)$ ,  $g \in \mathcal{G}$ , and  $\sup_{g \in \mathcal{G}} \|g\|_\infty$  is finite.

<sup>5</sup>Here, MSE is w.r.t.  $(\mathcal{X}_p, \mathcal{X}_u)$  for evaluating risks. Formally, it is over the probability measure  $\Pr(\mathcal{X}_p, \mathcal{X}_u)$ .

<sup>6</sup>This can be derived as  $n_p, n_u \rightarrow \infty$  by applying the *central limit theorem* to the two differences and then *L'Hôpital's rule* to the ratio of *complementary error functions* (Chung, 1968).

**Theorem 4** (Estimation error bound). *Assume that*

1.  $\inf_{g \in \mathcal{G}} R_u(g) \geq \alpha > 0$  and let  $\Delta$  be the right-hand side of (8);
2.  $\mathcal{G}$  is closed under negation, that is,  $g \in \mathcal{G}$  if and only if  $-g \in \mathcal{G}$ .

Then, for any  $\delta > 0$ , with probability at least  $1 - \delta$ ,

$$\begin{aligned} R(\tilde{g}_{pu}) - R(g^*) &\leq 16L_\ell \pi_p \mathfrak{R}_{n_p, p_p}(\mathcal{G}) \\ &\quad + 8L_\ell \mathfrak{R}_{n_u, p}(\mathcal{G}) + 2C'_\delta \cdot \chi_{n_p, n_u} + 2C_\ell \pi_p \Delta, \end{aligned} \quad (13)$$

where  $C'_\delta = C_\ell \sqrt{\ln(1/\delta)/2}$  and  $\mathfrak{R}_{n_p, p_p}(\mathcal{G})$  and  $\mathfrak{R}_{n_u, p}(\mathcal{G})$  are the Rademacher complexities of  $\mathcal{G}$  for the sampling of size  $n_p$  from  $p_p(x)$  and of size  $n_u$  from  $p(x)$ .

Theorem 4 ensures that learning by  $\tilde{R}_{pu}(g)$  is also consistent: As  $n_p$  and  $n_u$  approach infinity,  $R(\tilde{g}_{pu})$  converges to  $R(g^*)$ , and if  $\ell$  satisfies (6), all optimization problems are convex and  $\tilde{g}_{pu}$  converges to  $g^*$ . For many discriminative models, especially if  $g$  is linear in parameters and its norm is bounded,  $\mathfrak{R}_{n_p, p_p}(\mathcal{G})$  and  $\mathfrak{R}_{n_u, p}(\mathcal{G})$  decay in  $\mathcal{O}(1/\sqrt{n_p})$  and  $\mathcal{O}(1/\sqrt{n_u})$ , and the convergence rate from  $R(\tilde{g}_{pu})$  to  $R(g^*)$  is  $\mathcal{O}_p(\pi_p/\sqrt{n_p} + 1/\sqrt{n_u})$ .

For the sake of comparison, the estimation error  $R(\hat{g}_{pu}) - R(g^*)$  can be bounded as follows without the assumptions of Theorem 4:

$$\begin{aligned} R(\hat{g}_{pu}) - R(g^*) &\leq 8L_\ell \pi_p \mathfrak{R}_{n_p, p_p}(\mathcal{G}) \\ &\quad + 4L_\ell \mathfrak{R}_{n_u, p}(\mathcal{G}) + 2C_\delta \cdot \chi_{n_p, n_u}, \end{aligned} \quad (14)$$

where  $C_\delta = C_\ell \sqrt{\ln(2/\delta)/2}$ . We can see two main differences between (13) and (14):

- An extra term  $2C_\ell \pi_p \Delta$  is appended due to the bias of  $\tilde{R}_{pu}(g)$ . It is unessential since  $\Delta \approx \mathcal{O}(\exp(-n_p))$ ;
- Coefficients of the complexity terms are doubled due to the max in  $\tilde{R}_{pu}(g)$ . It is substantial since the rate is determined by the complexity terms and  $\chi_{n_p, n_u}$ .

Both differences come from the following uniform deviation bound of  $\tilde{R}_{pu}(g)$ , which is actually the whole foundation of Theorem 4.

**Lemma 5.** *Under the assumptions of Theorem 4, for any  $\delta > 0$ , with probability at least  $1 - \delta$ ,*

$$\begin{aligned} \sup_{g \in \mathcal{G}} |\tilde{R}_{pu}(g) - R(g)| &\leq 8L_\ell \pi_p \mathfrak{R}_{n_p, p_p}(\mathcal{G}) \\ &\quad + 4L_\ell \mathfrak{R}_{n_u, p}(\mathcal{G}) + C'_\delta \cdot \chi_{n_p, n_u} + C_\ell \pi_p \Delta. \end{aligned} \quad (15)$$

Note that  $\hat{R}_{pu}(g)$  is point-wise whereas  $\tilde{R}_{pu}(g)$  is not due to the max in it, making (15) much more difficult to prove than that of  $\hat{R}_{pu}(g)$ . The key trick is that after *symmetrization* we employ  $|\max\{0, z\} - \max\{0, z'\}| \leq |z - z'|$ , so that any difference of risks becomes point-wise. As a consequence, we have to play with an alternative definition of

Table 2. Specification of benchmark datasets.

Name	# Train	# Test	# Features	$\pi_p$
USPS	10,000	1,000	256	0.50
Adult	32,561	16,281	123	0.24
Web	49,749	14,951	300	0.03
MNIST	60,000	10,000	784	0.49

the Rademacher complexity with the absolute value inside the supremum, whose *contraction* leads to the doubled coefficients. We have to additionally assume that  $\mathcal{G}$  is closed under negation to change back to the original definition of the Rademacher complexity.

At a first glance, it seems  $\tilde{R}_{pu}(g)$  is worse than  $\hat{R}_{pu}(g)$  in terms of the estimation error bounds. In particular, similar bounds were used for theoretical comparisons of different risk estimators in Niu et al. (2016). Nevertheless, in order to be compared, the bounds in Niu et al. (2016) need to be proven using exactly the same technique. Since  $\tilde{R}_{pu}(g)$  is significantly different from  $\hat{R}_{pu}(g)$  for training  $g$  on some fixed data, by no means could we cope with  $\tilde{g}_{pu}$  using the same technique. Thus, it is futile to compare the tightness of estimation error bounds (13) and (14).

## 5. Experiments

In this section, we compare unbiased and non-negative PU learning by experiments.

### 5.1. Training linear models

Table 2 describes the specification of benchmark datasets. The datasets USPS and MNIST can be downloaded from the homepage of the late Sam Roweis;<sup>7</sup> The datasets Adult and Web can be downloaded from the website of LIBSVM.<sup>8</sup> USPS and MNIST were preprocessed in such a way that even digits form the P class and odd digits form the N class.

There were four learning methods involved:

- Oracle means PN learning where all training data are labeled. This indicates the best possible performance of the other three methods and it is only for reference but not for comparison;
- PN means PN learning where  $n_p = 1,000$  and  $n_n = n_p \cdot \min\{1, (\pi_n/2\pi_p)^2\}$ ;
- PU means unbiased PU learning where  $n_p = 1,000$  and  $n_u$  is the size of training data;
- NNPU means non-negative PU learning where  $n_p$  and  $n_u$  are same as PU.

<sup>7</sup>See <http://cs.nyu.edu/~roweis/data.html>.

<sup>8</sup>See <https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html>, where Adult is named a9a and Web is named w8a.

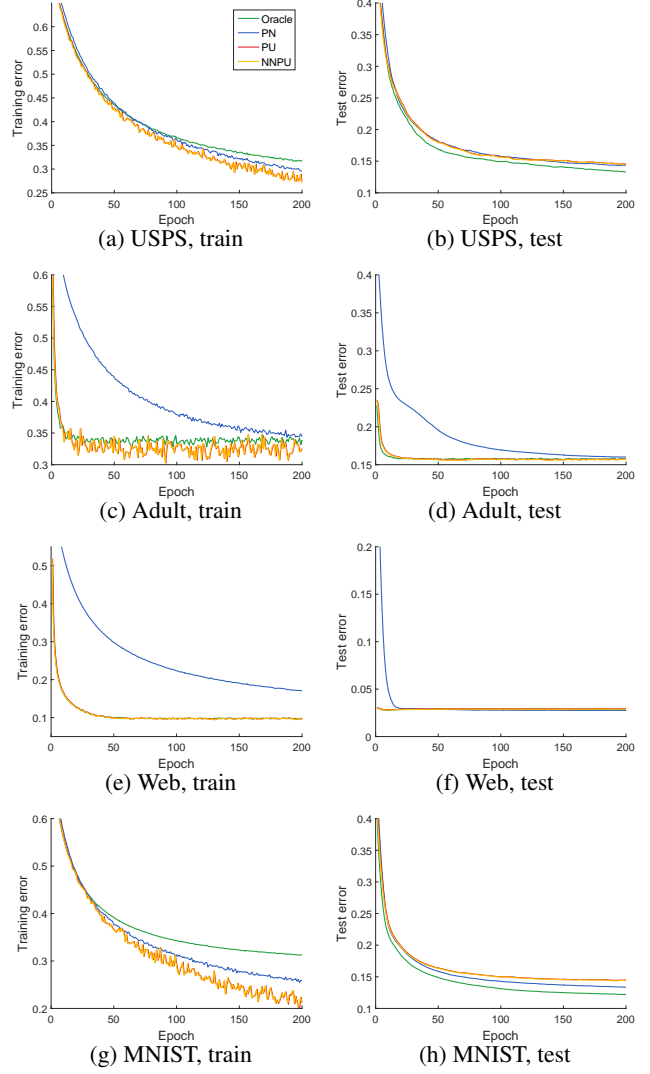


Figure 2. Experimental results of training linear models.

Note that for PU and NNPU learning, P and U data were not independent. An alternative is to use all the remaining training data as U data, but the difference is negligible.

The model being trained was  $g(x; w, b) = \langle w, x \rangle + b$ , where  $w \in \mathbb{R}^d$ ,  $b \in \mathbb{R}$  and  $\langle \cdot, \cdot \rangle$  denotes the inner product. The logistic loss was used and consequently the optimizations of all learning methods are convex. An additional  $\ell_2$ -regularization was appended to the objective functions, which were solved by Adam (Kingma & Ba, 2015). We set  $\beta = 2\sqrt{\pi_p \pi_n / n'_u}$  where  $n'_u$  is the number of U data in mini-batches and fix  $\gamma = 1$  for simplicity.

The experimental results are reported in Figure 2, where means of the training and test error based on ten random samplings are shown. We can see that if the model is simple, NNPU learning performed identically as PU learning. They were sometimes slightly worse than PN learning, and sometimes converged much faster to the oracle than PN learning.

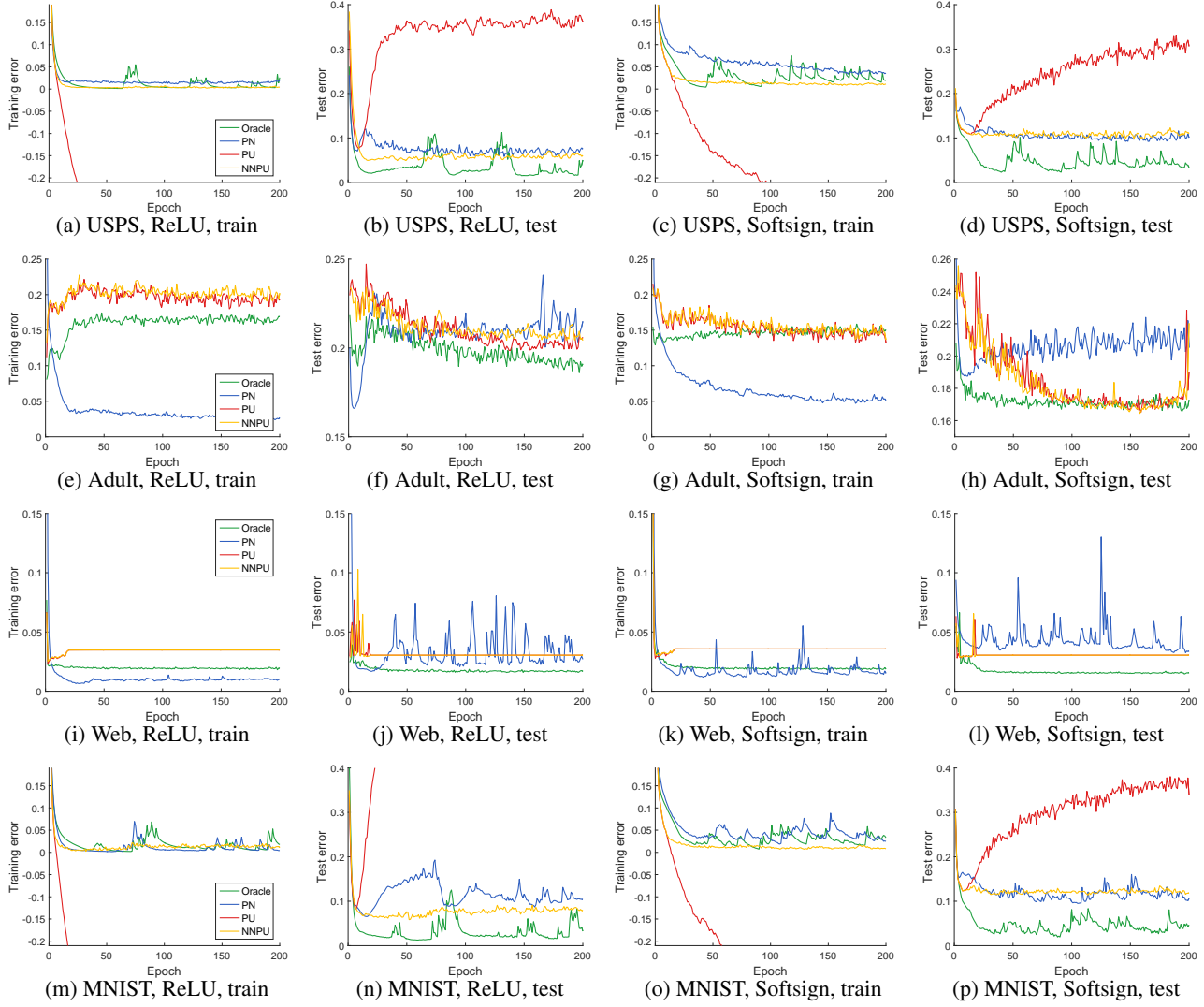


Figure 3. Experimental results of training deep neural networks.

## 5.2. Training deep neural networks

We also compared the four learning methods in training deep neural networks. The setup is very similar to the previous subsection. However, the model here was a fully-connected neural network with four hidden layers ( $d$ -300-300-300-1), the activation function was ReLU (Nair & Hinton, 2010) or Softsign (Glorot & Bengio, 2010), and batch normalization (Ioffe & Szegedy, 2015) was applied before all hidden layers. The sigmoid loss was used since it is bounded and it looks like the zero-one loss more than the logistic loss.

The experimental results are reported in Figure 3, where means of the training and test error based on the same ten random samplings are shown. We can see that on USPS and MNIST, PU learning overfitted training data a lot, in which cases NNPU learning improved on PU learning a lot. NNPU learning was also better than or comparable to PN

learning. On Adult and Web, PU learning did not overfit training data even though the model was very flexible. This is because the binary features of Adult are noisy and  $\pi_P$  of Web is too small. Instead, PN learning overfitted training data and became significantly worse than PU and NNPU learning. We can consider that NNPU learning successfully combines the advantages of PN and PU learning.

## 6. Conclusions

We proposed a non-negative risk estimator for PU learning that follows and improves on the state-of-the-art unbiased risk estimator. No matter how flexible the model is, it will not go negative like the unbiased counterparts. As a result, it is more robust against overfitting and training very flexible models given limited P data becomes possible. A large-scale PU learning algorithm was also developed. Extensive theoretical analyses are given as well.



A promising future direction is extending the current work to semi-supervised learning along the line of Sakai et al. (2016).

## References

- Bartlett, P. L. and Mendelson, S. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.
- Blanchard, G., Lee, G., and Scott, C. Semi-supervised novelty detection. *Journal of Machine Learning Research*, 11:2973–3009, 2010.
- Chung, K.-L. *A Course in Probability Theory*. Academic Press, 1968.
- De Comité, F., Denis, F., Gilleron, R., and Letouzey, F. Positive and unlabeled examples help learning. In *ALT*, 1999.
- Denis, F. PAC learning from positive statistical queries. In *ALT*, 1998.
- du Plessis, M. C., Niu, G., and Sugiyama, M. Analysis of learning from positive and unlabeled data. In *NIPS*, 2014.
- du Plessis, M. C., Niu, G., and Sugiyama, M. Convex formulation for learning from positive and unlabeled data. In *ICML*, 2015.
- du Plessis, M. C., Niu, G., and Sugiyama, M. Class-prior estimation for learning from positive and unlabeled data. *Machine Learning, to appear*, 2017.
- Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.
- Elkan, C. and Noto, K. Learning classifiers from only positive and unlabeled data. In *KDD*, 2008.
- Glorot, X. and Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, 2010.
- Halmos, P. R. *Measure Theory*. Springer-Verlag, 1974.
- Hsieh, C.-J., Natarajan, N., and Dhillon, I. S. PU learning for matrix completion. In *ICML*, 2015.
- Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.
- Kingma, D. P. and Ba, J. L. Adam: A method for stochastic optimization. In *ICLR*, 2015.
- Koltchinskii, V. Rademacher penalties and structural risk minimization. *IEEE Transactions on Information Theory*, 47(5): 1902–1914, 2001.
- Ledoux, M. and Talagrand, M. *Probability in Banach Spaces: Isoperimetry and Processes*. Springer, 1991.
- Lee, W. S. and Liu, B. Learning with positive and unlabeled examples using weighted logistic regression. In *ICML*, 2003.
- Letouzey, F., Denis, F., and Gilleron, R. Learning from positive and unlabeled examples. In *ALT*, 2000.
- Li, X. and Liu, B. Learning to classify texts using positive and unlabeled data. In *IJCAI*, 2003.
- Li, X., Yu, P. S., Liu, B., and Ng, S.-K. Positive unlabeled learning for data stream classification. In *SDM*, 2009.
- Liu, B., Lee, W. S., Yu, P. S., and Li, X. Partially supervised classification of text documents. In *ICML*, 2002.
- Liu, B., Dai, Y., Li, X., Lee, W. S., and Yu, P. S. Building text classifiers using positive and unlabeled examples. In *ICDM*, 2003.
- McDiarmid, C. On the method of bounded differences. In Siemons, J. (ed.), *Surveys in Combinatorics*, pp. 148–188. Cambridge University Press, 1989.
- Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of Machine Learning*. MIT Press, 2012.
- Nair, V. and Hinton, G. E. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010.
- Nguyen, M. N., Li, X., and Ng, S.-K. Positive unlabeled learning for time series classification. In *IJCAI*, 2011.
- Niu, G., du Plessis, M. C., Sakai, T., Ma, Y., and Sugiyama, M. Theoretical comparisons of positive-unlabeled learning against positive-negative learning. In *NIPS*, 2016.
- Patrini, G., Nielsen, F., Nock, R., and Carioni, M. Loss factorization, weakly supervised learning and label noise robustness. In *ICML*, 2016.
- Platt, J. C. Fast training of support vector machines using sequential minimal optimization. In Schölkopf, B., Burges, C. J. C., and Smola, A. J. (eds.), *Advances in Kernel Methods*, pp. 185–208. MIT Press, 1999.
- Ramaswamy, H. G., Scott, C., and Tewari, A. Mixture proportion estimation via kernel embedding of distributions. In *ICML*, 2016.
- Sakai, T., du Plessis, M. C., Niu, G., and Sugiyama, M. Semi-supervised classification based on classification from positive and unlabeled data. *arXiv preprint arXiv:1605.06955*, 2016.
- Sansone, E., De Natale, F. G. B., and Zhou, Z.-H. Efficient training for positive unlabeled learning. *arXiv preprint arXiv:1608.06807*, 2016.
- Scott, C. and Blanchard, G. Novelty detection: Unlabeled data definitely help. In *AISTATS*, 2009.
- Vapnik, V. N. *Statistical Learning Theory*. John Wiley & Sons, 1998.
- Ward, G., Hastie, T., Barry, S., Elith, J., and Leathwick, J. Presence-only data and the EM algorithm. *Biometrics*, 65(2): 554–563, 2009.
- Yuille, A. L. and Rangarajan, A. The concave-convex procedure (CCCP). In *NIPS*, 2001.

## A. Proofs

In this appendix, we prove all the theoretical results in Section 4.

### A.1. Proof of Lemma 1

Let  $p_p(\mathcal{X}_p) = p_p(x_1^p) \cdots p_p(x_{n_p}^p)$  and  $p(\mathcal{X}_u) = p(x_1^u) \cdots p(x_{n_u}^u)$  be the probability density functions of  $\mathcal{X}_p$  and  $\mathcal{X}_u$ . Then let  $F_p(\mathcal{X}_p)$  be the cumulative distribution function of  $\mathcal{X}_p$ ,  $F_u(\mathcal{X}_u)$  be that of  $\mathcal{X}_u$ , and  $F(\mathcal{X}_p, \mathcal{X}_u) = F_p(\mathcal{X}_p)F_u(\mathcal{X}_u)$  be the joint cumulative distribution function of  $(\mathcal{X}_p, \mathcal{X}_u)$ . Given these definitions, the measure of  $\mathfrak{D}^-(g)$  is defined by

$$\Pr(\mathfrak{D}^-(g)) = \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} dF(\mathcal{X}_p, \mathcal{X}_u),$$

where  $\Pr$  denotes the probability. Since  $\tilde{R}_{pu}(g)$  is identically  $\hat{R}_{pu}(g)$  on  $\mathfrak{D}^+(g)$  and different from  $\hat{R}_{pu}(g)$  on  $\mathfrak{D}^-(g)$ , we have  $\Pr(\mathfrak{D}^-(g)) = \Pr\{\tilde{R}_{pu}(g) \neq \hat{R}_{pu}(g)\}$ . That is, the measure of  $\mathfrak{D}^-(g)$  is non-zero if and only if  $\tilde{R}_{pu}(g)$  differs from  $\hat{R}_{pu}(g)$  with a non-zero probability.

Based on the facts that  $\hat{R}_{pu}(g)$  is unbiased and  $\tilde{R}_{pu}(g) - \hat{R}_{pu}(g) = 0$  on  $\mathfrak{D}^+(g)$ , we have

$$\begin{aligned} \mathbb{E}[\tilde{R}_{pu}(g)] - R(g) &= \mathbb{E}[\tilde{R}_{pu}(g) - \hat{R}_{pu}(g)] \\ &= \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^+(g)} (\tilde{R}_{pu}(g) - \hat{R}_{pu}(g)) dF(\mathcal{X}_p, \mathcal{X}_u) \\ &\quad + \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} (\tilde{R}_{pu}(g) - \hat{R}_{pu}(g)) dF(\mathcal{X}_p, \mathcal{X}_u) \\ &= \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} (\tilde{R}_{pu}(g) - \hat{R}_{pu}(g)) dF(\mathcal{X}_p, \mathcal{X}_u). \end{aligned}$$

As a result,  $\mathbb{E}[\tilde{R}_{pu}(g)] - R(g) > 0$  if and only if  $\int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} dF(\mathcal{X}_p, \mathcal{X}_u) > 0$  due to the fact  $\tilde{R}_{pu}(g) - \hat{R}_{pu}(g) > 0$  on  $\mathfrak{D}^-(g)$ . That is, the bias of  $\tilde{R}_{pu}(g)$  is positive if and only if the measure of  $\mathfrak{D}^-(g)$  is non-zero.

We prove (8) by the method of bounded differences, for that

$$\mathbb{E}[\hat{R}_u^-(g) - \pi_p \hat{R}_p^-(g)] = R_u^-(g) - \pi_p R_p^-(g) = R_n(g) \geq \alpha.$$

We have assumed that  $0 \leq \ell(t, \pm 1) \leq C_\ell$ , and thus the change of  $\hat{R}_p^-(g)$  will be no more than  $C_\ell/n_p$  if some  $x_i^p \in \mathcal{X}_p$  is replaced, or the change of  $\hat{R}_u^-(g)$  will be no more than  $C_\ell/n_u$  if some  $x_i^u \in \mathcal{X}_u$  is replaced. Subsequently, *McDiarmid's inequality* (McDiarmid, 1989) implies

$$\begin{aligned} \Pr\{R_n(g) - (\hat{R}_u^-(g) - \pi_p \hat{R}_p^-(g)) \geq \alpha\} &\leq \exp\left(-\frac{2\alpha^2}{n_p(C_\ell\pi_p/n_p)^2 + n_u(C_\ell/n_u)^2}\right) \\ &= \exp\left(-\frac{2\alpha^2/C_\ell^2}{\pi_p^2/n_p + 1/n_u}\right). \end{aligned}$$

Taking into account that

$$\begin{aligned} \Pr(\mathfrak{D}^-(g)) &= \Pr\{\hat{R}_u^-(g) - \pi_p \hat{R}_p^-(g) < 0\} \\ &\leq \Pr\{\hat{R}_u^-(g) - \pi_p \hat{R}_p^-(g) \leq R_n(g) - \alpha\} \\ &= \Pr\{R_n(g) - (\hat{R}_u^-(g) - \pi_p \hat{R}_p^-(g)) \geq \alpha\}, \end{aligned}$$

we complete the proof.  $\square$

### A.2. Proof of Theorem 2

It has been proven in Lemma 1 that  $\mathbb{E}[\tilde{R}_{\text{pu}}(g)] - R(g) = \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} (\tilde{R}_{\text{pu}}(g) - \hat{R}_{\text{pu}}(g)) dF(\mathcal{X}_p, \mathcal{X}_u)$ , and thus the exponential decay of the bias in (9) is obtained via

$$\begin{aligned} \mathbb{E}[\tilde{R}_{\text{pu}}(g)] - R(g) &\leq \sup_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} (\tilde{R}_{\text{pu}}(g) - \hat{R}_{\text{pu}}(g)) \cdot \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} dF(\mathcal{X}_p, \mathcal{X}_u) \\ &= \sup_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} (\pi_p \hat{R}_p^-(g) - \hat{R}_u^-(g)) \cdot \Pr(\mathfrak{D}^-(g)) \\ &\leq C_\ell \pi_p \Delta_g. \end{aligned}$$

The deviation bound (10) is due to

$$\begin{aligned} |\tilde{R}_{\text{pu}}(g) - R(g)| &\leq |\tilde{R}_{\text{pu}}(g) - \mathbb{E}[\tilde{R}_{\text{pu}}(g)]| + |\mathbb{E}[\tilde{R}_{\text{pu}}(g)] - R(g)| \\ &\leq |\tilde{R}_{\text{pu}}(g) - \mathbb{E}[\tilde{R}_{\text{pu}}(g)]| + C_\ell \pi_p \Delta_g. \end{aligned}$$

The change of  $\tilde{R}_{\text{pu}}(g)$  will be no more than  $2C_\ell/n_p$  if some  $x_i^p \in \mathcal{X}_p$  is replaced, or it will be no more than  $C_\ell/n_u$  if some  $x_i^u \in \mathcal{X}_u$  is replaced, and McDiarmid's inequality gives us

$$\Pr\{|\tilde{R}_{\text{pu}}(g) - \mathbb{E}[\tilde{R}_{\text{pu}}(g)]| \geq \epsilon\} \leq 2 \exp\left(-\frac{2\epsilon^2}{n_p(2C_\ell\pi_p/n_p)^2 + n_u(C_\ell/n_u)^2}\right),$$

or equivalently, with probability at least  $1 - \delta$ ,

$$\begin{aligned} |\tilde{R}_{\text{pu}}(g) - \mathbb{E}[\tilde{R}_{\text{pu}}(g)]| &\leq \sqrt{\frac{\ln(2/\delta)C_\ell^2}{2} \left(\frac{4\pi_p^2}{n_p} + \frac{1}{n_u}\right)} \\ &\leq C_\delta \left(\frac{2\pi_p}{\sqrt{n_p}} + \frac{1}{\sqrt{n_u}}\right) \\ &= C_\delta \cdot \chi_{n_p, n_u}. \end{aligned}$$

On the other hand, the deviation bound (11) is due to

$$|\tilde{R}_{\text{pu}}(g) - R(g)| \leq |\tilde{R}_{\text{pu}}(g) - \hat{R}_{\text{pu}}(g)| + |\hat{R}_{\text{pu}}(g) - R(g)|,$$

where  $|\tilde{R}_{\text{pu}}(g) - \hat{R}_{\text{pu}}(g)| > 0$  with probability at most  $\Delta_g$  and  $|\hat{R}_{\text{pu}}(g) - R(g)|$  shares the same concentration inequality with  $|\tilde{R}_{\text{pu}}(g) - \mathbb{E}[\tilde{R}_{\text{pu}}(g)]|$ .  $\square$

### A.3. Proof of Theorem 3

For convenience, let  $A = \pi_p \hat{R}_p^+(g)$  and  $B = \hat{R}_u^-(g) - \pi_p \hat{R}_p^-(g)$ , so that  $\hat{R}_{\text{pu}}(g) = A + B$  and  $\tilde{R}_{\text{pu}}(g) = A + B_+$  where  $B_+ = \max\{0, B\}$ . Subsequently,

$$\begin{aligned} \text{MSE}(\hat{R}_{\text{pu}}(g)) &= \mathbb{E}[(A + B - \mathbb{E}[A + B])^2] \\ &= \mathbb{E}[(A + B)^2] - 2\mathbb{E}[A + B] \cdot \mathbb{E}[A + B] + (\mathbb{E}[A + B])^2, \\ \text{MSE}(\tilde{R}_{\text{pu}}(g)) &= \mathbb{E}[(A + B_+ - \mathbb{E}[A + B])^2] \\ &= \mathbb{E}[(A + B_+)^2] - 2\mathbb{E}[A + B] \cdot \mathbb{E}[A + B_+] + (\mathbb{E}[A + B])^2. \end{aligned}$$

Hence,

$$\text{MSE}(\hat{R}_{\text{pu}}(g)) - \text{MSE}(\tilde{R}_{\text{pu}}(g)) = \{\mathbb{E}[(A + B)^2] - \mathbb{E}[(A + B_+)^2]\} - \{2\mathbb{E}[A + B] \cdot (\mathbb{E}[A + B] - \mathbb{E}[A + B_+])\}.$$

The first term can be rewritten as

$$\begin{aligned} \mathbb{E}[(A + B)^2] - \mathbb{E}[(A + B_+)^2] &= \mathbb{E}[2A(B - B_+) + B^2 - B_+^2] \\ &= \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} (2AB + B^2) dF(\mathcal{X}_p, \mathcal{X}_u) \\ &= 2 \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} AB dF(\mathcal{X}_p, \mathcal{X}_u) + \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} B^2 dF(\mathcal{X}_p, \mathcal{X}_u), \end{aligned}$$

and the second term can be rewritten as

$$\begin{aligned} 2\mathbb{E}[A + B] \cdot (\mathbb{E}[A + B] - \mathbb{E}[A + B_+]) &= 2\mathbb{E}[A + B] \cdot \mathbb{E}[B - B_+] \\ &= 2 \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} (\mathbb{E}[A] + \mathbb{E}[B]) B dF(\mathcal{X}_p, \mathcal{X}_u). \end{aligned}$$

As a consequence,

$$\begin{aligned} \text{MSE}(\hat{R}_{\text{pu}}(g)) - \text{MSE}(\tilde{R}_{\text{pu}}(g)) &= 2 \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} (A - \mathbb{E}[A] - \mathbb{E}[B]) B dF(\mathcal{X}_p, \mathcal{X}_u) \\ &\quad + \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} B^2 dF(\mathcal{X}_p, \mathcal{X}_u). \end{aligned}$$

Therefore, in order to prove the theorem, it suffices to show that  $A - \mathbb{E}[A] - \mathbb{E}[B] \leq B$  on  $\mathfrak{D}^-(g)$ .

By the assumption that  $\ell$  satisfies (4),

$$\begin{aligned} A - \mathbb{E}[A] - \mathbb{E}[B] &= \pi_p \hat{R}_p^+(g) - \pi_p R_p^+(g) - \mathbb{E}[B] \\ &= \pi_p R_p^-(g) - \pi_p \hat{R}_p^-(g) - \mathbb{E}[B]. \end{aligned}$$

Thus, with probability one,

$$\begin{aligned} A - \mathbb{E}[A] - \mathbb{E}[B] &= \pi_p R_p^-(g) - \pi_p \hat{R}_p^-(g) - \mathbb{E}[B] + \hat{R}_u^-(g) - \hat{R}_u^-(g) + R_u^-(g) - R_u^-(g) \\ &= (\hat{R}_u^-(g) - \pi_p \hat{R}_p^-(g)) - (R_u^-(g) - \pi_p R_p^-(g)) - \mathbb{E}[B] + (R_u^-(g) - \hat{R}_u^-(g)) \\ &= B - 2\mathbb{E}[B] + (R_u^-(g) - \hat{R}_u^-(g)) \\ &\leq B, \end{aligned}$$

where we used the assumptions that  $\mathbb{E}[B] \geq \alpha$  and  $R_u^-(g) - \hat{R}_u^-(g) \leq 2\alpha$  almost surely on  $\mathfrak{D}^-(g)$ .

To sum up, we have established that

$$\text{MSE}(\hat{R}_{\text{pu}}(g)) - \text{MSE}(\tilde{R}_{\text{pu}}(g)) \geq 3 \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} B^2 dF(\mathcal{X}_p, \mathcal{X}_u).$$

Since  $B^2 > 0$  on  $\mathfrak{D}^-(g)$  and  $\Pr(\mathfrak{D}^-(g)) > 0$ , we obtain  $\text{MSE}(\hat{R}_{\text{pu}}(g)) - \text{MSE}(\tilde{R}_{\text{pu}}(g)) > 0$ . Finally, for any  $0 \leq \beta \leq C_\ell \pi_p$ , it is clear that  $\{(\mathcal{X}_p, \mathcal{X}_u) \mid B < -\beta\} \subseteq \mathfrak{D}^-(g)$  and  $B < -\beta$  if and only if  $\tilde{R}_{\text{pu}}(g) - \hat{R}_{\text{pu}}(g) > \beta$ . These two facts mean that

$$\begin{aligned} \int_{(\mathcal{X}_p, \mathcal{X}_u) \in \mathfrak{D}^-(g)} B^2 dF(\mathcal{X}_p, \mathcal{X}_u) &\geq \int_{(\mathcal{X}_p, \mathcal{X}_u) \mid B < -\beta} B^2 dF(\mathcal{X}_p, \mathcal{X}_u) \\ &\geq \beta^2 \int_{(\mathcal{X}_p, \mathcal{X}_u) \mid B < -\beta} dF(\mathcal{X}_p, \mathcal{X}_u) \\ &= \beta^2 \Pr\{B < -\beta\} \\ &= \beta^2 \Pr\{\tilde{R}_{\text{pu}}(g) - \hat{R}_{\text{pu}}(g) > \beta\}, \end{aligned}$$

which proves (12) and the whole theorem.  $\square$

#### A.4. Proof of Lemma 5

**Preliminary** An alternative definition of the Rademacher complexity will be used during the proof:

$$\mathfrak{R}'_{n,q}(\mathcal{G}) = \mathbb{E}_{\mathcal{X}} \mathbb{E}_{\sigma_1, \dots, \sigma_n} \left[ \sup_{g \in \mathcal{G}} \left| \frac{1}{n} \sum_{x_i \in \mathcal{X}} \sigma_i g(x_i) \right| \right].$$

For the sake of comparison, the one we have used in the statements of theoretical results is

$$\mathfrak{R}_{n,q}(\mathcal{G}) = \mathbb{E}_{\mathcal{X}} \mathbb{E}_{\sigma_1, \dots, \sigma_n} \left[ \sup_{g \in \mathcal{G}} \frac{1}{n} \sum_{x_i \in \mathcal{X}} \sigma_i g(x_i) \right].$$

This alternative version comes from [Koltchinskii \(2001\)](#); [Bartlett & Mendelson \(2002\)](#) of which authors are the pioneers of error bounds based on the Rademacher complexity. Without any composition,  $\mathfrak{R}'_{n,q}(\mathcal{G}) \geq \mathfrak{R}_{n,q}(\mathcal{G})$  for arbitrary  $\mathcal{G}$  and  $\mathfrak{R}'_{n,q}(\mathcal{G}) = \mathfrak{R}_{n,q}(\mathcal{G})$  if  $\mathcal{G}$  is closed under negation. However, with a composition  $\ell \circ \mathcal{G} = \{\ell \circ g \mid g \in \mathcal{G}\}$  where the loss  $\ell$  is non-negative, the Rademacher complexity of the *composite function class* will meet  $\mathfrak{R}'_{n,q}(\ell \circ \mathcal{G}) \geq \mathfrak{R}_{n,q}(\ell \circ \mathcal{G})$  for any  $\mathcal{G}$ . Further, a vital disagreement arises when considering the so-called contraction principle or property: If  $\psi : \mathbb{R} \rightarrow \mathbb{R}$  is a Lipschitz continuous function with a Lipschitz constant  $L_\psi$  and satisfies  $\psi(0) = 0$ , we have

$$\begin{aligned} \mathfrak{R}_{n,q}(\psi \circ \mathcal{G}) &\leq L_\psi \mathfrak{R}_{n,q}(\mathcal{G}), \\ \mathfrak{R}'_{n,q}(\psi \circ \mathcal{G}) &\leq 2L_\psi \mathfrak{R}'_{n,q}(\mathcal{G}), \end{aligned}$$

according to *Talagrand's contraction lemma* ([Ledoux & Talagrand, 1991](#)).

**Proof** Firstly, we deal with the bias of  $\tilde{R}_{\text{pu}}(g)$ :

$$\begin{aligned} \sup_{g \in \mathcal{G}} |\tilde{R}_{\text{pu}}(g) - R(g)| &\leq \sup_{g \in \mathcal{G}} |\tilde{R}_{\text{pu}}(g) - \mathbb{E}[\tilde{R}_{\text{pu}}(g)]| + \sup_{g \in \mathcal{G}} |\mathbb{E}[\tilde{R}_{\text{pu}}(g)] - R(g)| \\ &\leq \sup_{g \in \mathcal{G}} |\tilde{R}_{\text{pu}}(g) - \mathbb{E}[\tilde{R}_{\text{pu}}(g)]| + C_\ell \pi_p \Delta, \end{aligned} \quad (16)$$

where we followed the assumption that  $\inf_{g \in \mathcal{G}} R_n(g) \geq \alpha > 0$  and Theorem 2.

Secondly, we apply McDiarmid's inequality to the uniform deviation  $\sup_{g \in \mathcal{G}} |\tilde{R}_{\text{pu}}(g) - \mathbb{E}[\tilde{R}_{\text{pu}}(g)]|$  to get that with probability at least  $1 - \delta$ ,

$$\sup_{g \in \mathcal{G}} |\tilde{R}_{\text{pu}}(g) - \mathbb{E}[\tilde{R}_{\text{pu}}(g)]| \leq C'_\delta \cdot \chi_{n_p, n_u}. \quad (17)$$

Notice that this concentration inequality is single-sided even though the uniform deviation itself is double-sided, which is different from the non-uniform deviation in Theorem 2.

Thirdly, we make *symmetrization* ([Vapnik, 1998](#)). Suppose that  $(\mathcal{X}'_p, \mathcal{X}'_u)$  is a *ghost sample* for symmetrization, then

$$\begin{aligned} \mathbb{E}[\sup_{g \in \mathcal{G}} |\tilde{R}_{\text{pu}}(g) - \mathbb{E}[\tilde{R}_{\text{pu}}(g)]|] &= \mathbb{E}_{(\mathcal{X}_p, \mathcal{X}_u)} [\sup_{g \in \mathcal{G}} |\tilde{R}_{\text{pu}}(g) - \mathbb{E}_{(\mathcal{X}'_p, \mathcal{X}'_u)} [\tilde{R}_{\text{pu}}(g)]|] \\ &\leq \mathbb{E}_{(\mathcal{X}_p, \mathcal{X}_u), (\mathcal{X}'_p, \mathcal{X}'_u)} [\sup_{g \in \mathcal{G}} |\tilde{R}_{\text{pu}}(g; \mathcal{X}_p, \mathcal{X}_u) - \tilde{R}_{\text{pu}}(g; \mathcal{X}'_p, \mathcal{X}'_u)|], \end{aligned}$$

where we applied *Jensen's inequality* twice since the absolute value and the supremum are both convex. By decomposing the difference  $|\tilde{R}_{\text{pu}}(g; \mathcal{X}_p, \mathcal{X}_u) - \tilde{R}_{\text{pu}}(g; \mathcal{X}'_p, \mathcal{X}'_u)|$ , we can know that

$$\begin{aligned} &|\tilde{R}_{\text{pu}}(g; \mathcal{X}_p, \mathcal{X}_u) - \tilde{R}_{\text{pu}}(g; \mathcal{X}'_p, \mathcal{X}'_u)| \\ &= |\pi_p \hat{R}_p^+(g; \mathcal{X}_p) - \pi_p \hat{R}_p^+(g; \mathcal{X}'_p)| \\ &\quad + \max\{0, \hat{R}_u^-(g; \mathcal{X}_u) - \pi_p \hat{R}_p^-(g; \mathcal{X}_p)\} - \max\{0, \hat{R}_u^-(g; \mathcal{X}'_u) - \pi_p \hat{R}_p^-(g; \mathcal{X}'_p)\} \\ &\leq \pi_p |\hat{R}_p^+(g; \mathcal{X}_p) - \hat{R}_p^+(g; \mathcal{X}'_p)| + \pi_p |\hat{R}_p^-(g; \mathcal{X}_p) - \hat{R}_p^-(g; \mathcal{X}'_p)| + |\hat{R}_u^-(g; \mathcal{X}_u) - \hat{R}_u^-(g; \mathcal{X}'_u)| \end{aligned}$$

where we employed  $|\max\{0, z\} - \max\{0, z'\}| \leq |z - z'|$ . This decomposition results in

$$\begin{aligned} \mathbb{E}[\sup_{g \in \mathcal{G}} |\tilde{R}_{\text{pu}}(g) - \mathbb{E}[\tilde{R}_{\text{pu}}(g)]|] &\leq \pi_p \mathbb{E}_{\mathcal{X}_p, \mathcal{X}'_p} [\sup_{g \in \mathcal{G}} |\hat{R}_p^+(g; \mathcal{X}_p) - \hat{R}_p^+(g; \mathcal{X}'_p)|] \\ &\quad + \pi_p \mathbb{E}_{\mathcal{X}_p, \mathcal{X}'_p} [\sup_{g \in \mathcal{G}} |\hat{R}_p^-(g; \mathcal{X}_p) - \hat{R}_p^-(g; \mathcal{X}'_p)|] \\ &\quad + \mathbb{E}_{\mathcal{X}_u, \mathcal{X}'_u} [\sup_{g \in \mathcal{G}} |\hat{R}_u^-(g; \mathcal{X}_u) - \hat{R}_u^-(g; \mathcal{X}'_u)|]. \end{aligned} \quad (18)$$

Fourthly, we relax the expectations in (18) to Rademacher complexities. The original surrogate loss  $\ell$  may miss the origin, i.e.,  $\ell(0, y) \neq 0$ , with which we need to deal. Let  $\tilde{\ell}(t, y) = \ell(t, y) - \ell(0, y)$  be a *shifted loss* so that  $\tilde{\ell}(0, y) = 0$ . Note that



for all  $t, t' \in \mathbb{R}$  and  $y = \pm 1$ ,  $\ell(t, y) - \ell(t', y) = \tilde{\ell}(t, y) - \tilde{\ell}(t', y)$ . Hence,

$$\begin{aligned}\hat{R}_p^+(g; \mathcal{X}_p) - \hat{R}_p^+(g; \mathcal{X}'_p) &= (1/n_p) \sum_{x_i \in \mathcal{X}_p} \ell(g(x_i), +1) - (1/n_p) \sum_{x'_i \in \mathcal{X}'_p} \ell(g(x'_i), +1) \\ &= (1/n_p) \sum_{i=1}^{n_p} (\ell(g(x_i), +1) - \ell(g(x'_i), +1)) \\ &= (1/n_p) \sum_{i=1}^{n_p} (\tilde{\ell}(g(x_i), +1) - \tilde{\ell}(g(x'_i), +1)).\end{aligned}$$

This is already a standard form that we can attach Rademacher variables to every  $\tilde{\ell}(g(x_i), +1) - \tilde{\ell}(g(x'_i), +1)$ , and it is a routine work to show that

$$\mathbb{E}_{\mathcal{X}_p, \mathcal{X}'_p} [\sup_{g \in \mathcal{G}} |\hat{R}_p^+(g; \mathcal{X}_p) - \hat{R}_p^+(g; \mathcal{X}'_p)|] \leq 2\mathfrak{R}_{n_p, p_p}(\tilde{\ell}(\cdot, +1) \circ \mathcal{G}).$$

The other two expectations can be handled analogously. As a result, (18) can be reduced to

$$\mathbb{E}[\sup_{g \in \mathcal{G}} |\tilde{R}_{pu}(g) - \mathbb{E}[\tilde{R}_{pu}(g)]|] \leq 2\pi_p \mathfrak{R}'_{n_p, p_p}(\tilde{\ell}(\cdot, +1) \circ \mathcal{G}) + 2\pi_p \mathfrak{R}'_{n_p, p_p}(\tilde{\ell}(\cdot, -1) \circ \mathcal{G}) + 2\mathfrak{R}'_{n_u, p}(\tilde{\ell}(\cdot, -1) \circ \mathcal{G}). \quad (19)$$

Finally, we contract the Rademacher complexities of composite function classes to those of the original function class. It is obvious that  $\tilde{\ell}$  shares the same Lipschitz constant  $L_\ell$  with  $\ell$ , and consequently

$$\begin{aligned}\mathfrak{R}'_{n_p, p_p}(\tilde{\ell}(\cdot, +1) \circ \mathcal{G}) &\leq 2L_\ell \mathfrak{R}'_{n_p, p_p}(\mathcal{G}) = 2L_\ell \mathfrak{R}_{n_p, p_p}(\mathcal{G}) \\ \mathfrak{R}'_{n_p, p_p}(\tilde{\ell}(\cdot, -1) \circ \mathcal{G}) &\leq 2L_\ell \mathfrak{R}'_{n_p, p_p}(\mathcal{G}) = 2L_\ell \mathfrak{R}_{n_p, p_p}(\mathcal{G}) \\ \mathfrak{R}'_{n_u, p}(\tilde{\ell}(\cdot, -1) \circ \mathcal{G}) &\leq 2L_\ell \mathfrak{R}'_{n_u, p}(\mathcal{G}) = 2L_\ell \mathfrak{R}_{n_u, p}(\mathcal{G}),\end{aligned} \quad (20)$$

where we used Talagrand's contraction lemma and the assumption that  $\mathcal{G}$  is closed under negation. Combining (16), (17), (19) and (20) finishes the proof of the uniform deviation bound (15).  $\square$

#### A.5. Proof of Theorem 4

Based on Lemma 5, the estimation error bound (13) is proven through

$$\begin{aligned}R(\tilde{g}_{pu}) - R(g^*) &= \left( \tilde{R}_{pu}(\tilde{g}_{pu}) - \tilde{R}_{pu}(g^*) \right) + \left( R(\tilde{g}_{pu}) - \tilde{R}_{pu}(\tilde{g}_{pu}) \right) + \left( \tilde{R}_{pu}(g^*) - R(g^*) \right) \\ &\leq 0 + 2 \sup_{g \in \mathcal{G}} |\tilde{R}_{pu}(g) - R(g)| \\ &\leq 16L_\ell \pi_p \mathfrak{R}_{n_p, p_p}(\mathcal{G}) + 8L_\ell \mathfrak{R}_{n_u, p}(\mathcal{G}) + 2C'_\delta \cdot \chi_{n_p, n_u} + 2C_\ell \pi_p \Delta,\end{aligned}$$

where  $\tilde{R}_{pu}(\tilde{g}_{pu}) \leq \tilde{R}_{pu}(g^*)$  by the definition of  $\tilde{g}_{pu}$ .  $\square$